

Using Scripts (Macros) in MIPAV

MIPAV provides three different methods of customizing the program. The first method involves developing scripts, which you use directly within the program. In the second method, you also develop scripts but you initiate, or call, them from another program. Developing scripts does not require programming skills or learning a new script language, and calling them from another program may require, depending on the program, only minimal programming knowledge. This chapter discusses developing and using scripts.

The third method of customizing MIPAV—developing plug-in programs—does require Java programming skills. For more information about this method, refer to [Chapter 11, “Developing Plug-in Programs.”](#) for more information.

Using scripts

Scripts, sometimes referred to as *macros* in other programs, record a series of commands or actions on specific images or groups of images that you can run with a single command. Using scripts can increase productivity and improve efficiency in performing commonly repeated actions or a series of actions. Most important, you can use scripts to process a large set of user-defined images.

Displaying the scripting toolbar

To display the scripting toolbar, select **Toolbar > Scripting** in the MIPAV window. The scripting toolbar immediately appears beneath the VOI toolbar or, if you choose to also display the Paint toolbar, immediately below that.



Remember: Select **Help > Program Options** in the main MIPAV window and then select **Show Scripting toolbar** in the MIPAV Options dialog box to always display the scripting toolbar.

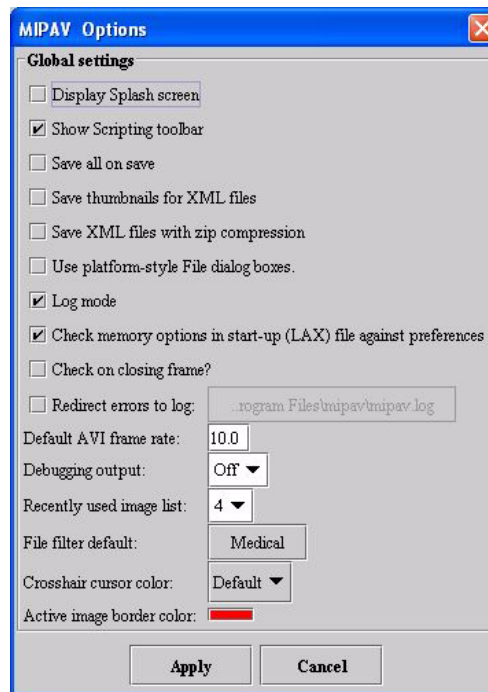


Figure 269. MIPAV Options dialog box with Show scripting toolbar selected

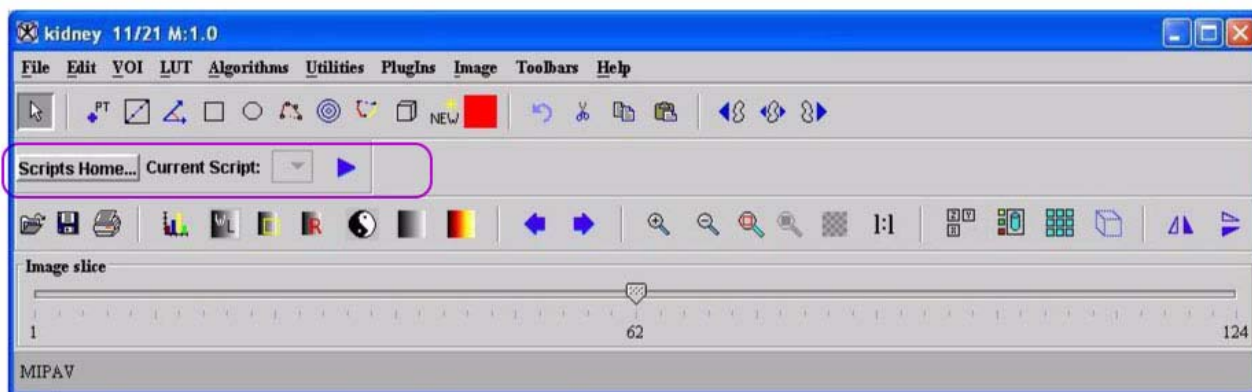
Creating scripts

The *scripts home* is the directory in which you store scripts and from which you run them. After you select a scripts home, the name of one of the scripts (which are ordered alphabetically) in that directory appears after *Current Script* on the scripting toolbar (Figure 270). If there are several scripts in

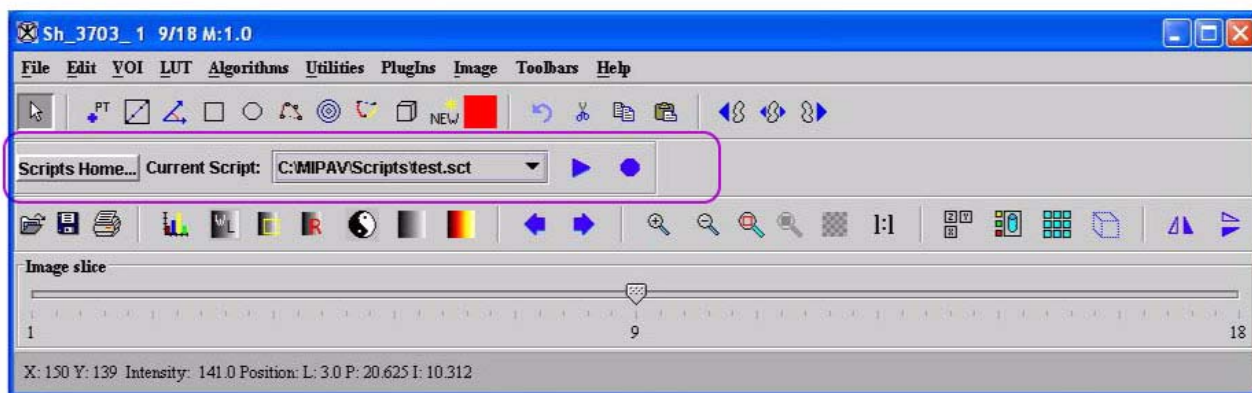
the directory, Current Script becomes a list box from which you can select the script you want to run.



Recommendation: Although you can select or change the scripts home at any time, you might want to do so before creating any scripts to complete the setup of your system.




(A) The MIPAV window before selecting a scripts home



(B) The MIPAV window after selecting a scripts home and running a script for the first time

Figure 270. The MIPAV window (A) before selecting a scripts home and (B) after selecting a scripts home and running the first script

Before creating scripts, it is helpful to understand that there are two methods for creating them:

- *Active mode*—In active mode one or more images are already open when you create the script. Because images are open, MIPAV assumes that the script applies to the active image. The *active image* is the image that is currently selected (Figure 271). To create a script in active mode, you simply click , the Start Recording Script icon, on the scripting toolbar.
- *Group mode*—In group mode no images are open when you begin creating the script, and the MIPAV window appears in its initial small size, which only displays some of the menus. Generally, you use the group mode to create scripts that apply to multiple image files. To create a script in group mode, click File > Scripts > Record Script.

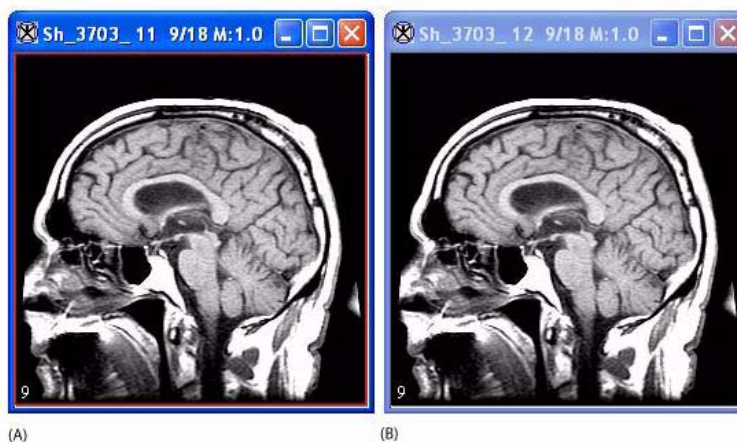


Figure 271. An (A) active image, whose title bar is highlighted, and (B) inactive image, whose title bar is dimmed

It's also important to understand which actions you can record in a script. For example, scripts do not record views, such as lightbox, triplanar, and volume renderer nor do they record other visualization commands, such as those that adjust the lookup table. Actions that you can record in a script include all of the algorithms and utility commands and the creation of VOIs.

This section explains the following tasks:

- Selecting a scripts home (see “To select a scripts home” on page 396)
- Creating scripts in active mode (see “To create scripts using the active mode” on page 397)
- Creating scripts in group mode (“To create scripts using the group mode” on page 400)
- Stop recording scripts during creation (“To stop recording scripts during creation” on page 402)

To select a scripts home

- 1** Start MIPAV. The initial MIPAV window opens.
- 2** Open an image. The expanded MIPAV window appears.
- 3** Select Toolbars > Scripting if the scripting toolbar is not displayed. The scripting toolbar appears (Figure 272A).



Figure 272. A comparison of (A) the initial scripting toolbar and (B) the toolbar as it looks after you identify a scripts home and run your first script



Note: Notice that a dimmed rectangle appears immediately following the words *Current Script* in the scripting toolbar. After you select a scripts home and run a script for the first time, Current Script becomes a list box that shows the name of a script in the scripts home directory (Figure 272B).

- 4** Select Scripts Home. The Choose Directory dialog box (Figure 273) appears.



Note: The Choose Directory dialog box only shows directories, not individual files.

- 5 Select a directory in which to store your scripts.
- 6 Click Open. The Choose Directory dialog box closes.

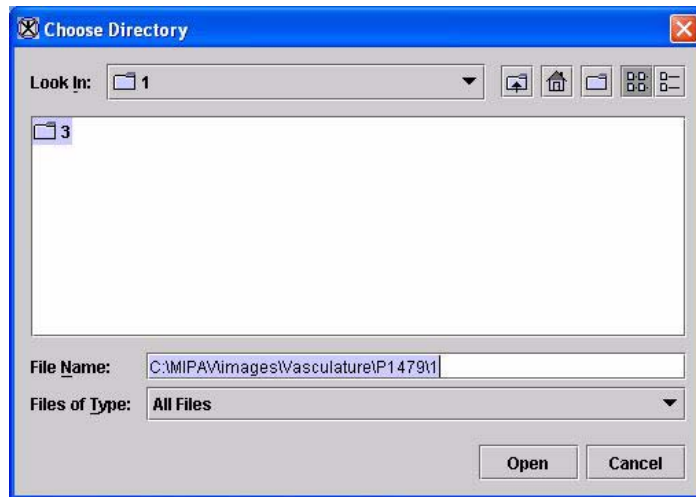



Figure 273. Choose Directory dialog box

To create scripts using the active mode

- 1 Start MIPAV. The initial MIPAV window and the Output window open.
- 2 Open the image on which you want to run the script. The MIPAV window expands.
- 3 Click , the Start recording script icon, on the scripting toolbar to create a script in active mode.



Note: The icon turns red () to indicate that MIPAV is now recording a script.

The Record New Script dialog box (Figure 275) opens. The following appears at the top of the dialog box: “The script is now recording. Your actions will be recorded below.”

File	Open —Opens a previously saved script. When you select this command, the Open dialog box appears.
	Save —Saves the script under the name you specify. When you select this command, the Save dialog box appears.
	Exit —Closes this dialog box without saving the script.
Pause	Stops recording the script. When you select this button, the name of the button changes to "Resume."
Resume	Activates the recording process. When you select this button, the name changes to "Pause."
Scripting box	Displays the actions in the script as a series of commands.
Enable Edit	Allows you to make changes or corrections to the script. When you select this button, the scripting box turns from gray to white to indicate that you can now type, copy, or past information into it or delete information from it.
Disable Edit	Prevents any changes or corrections being made to the existing script. When you select this button, the scripting box turns from white to gray and the button name changes to "Enable Edit."

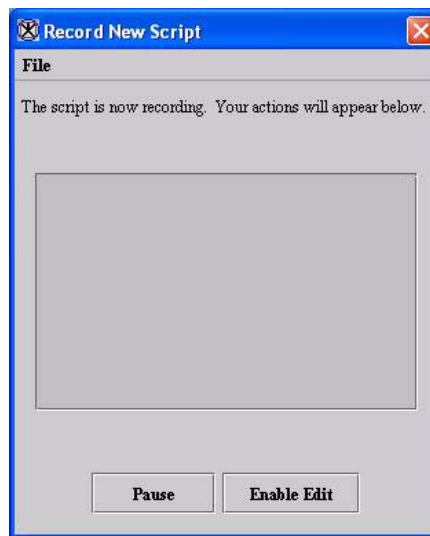


Figure 274. Record New Script dialog box

- 4 Delineate VOIs and perform as many actions (algorithms and utilities) as you want on the image. VOI commands do not appear on the scripting box on the Record New Script dialog box. To run the script later, you need to save the VOIs.

If you need to create a VOI to perform an action, save the VOI by selecting VOI > Save VOI. The program saves the VOI to the Scripts Home directory under the name *AreaN.voi* where *N* is a number that is assigned sequentially. The name of the first VOI that you save is *Area1.voi*; the second is *Area2.voi*; and so on. Make sure you type the .voi extension to the file.

If you want to specify a different name for the VOI, select VOI > Save VOI as. The Save VOI as dialog box opens. Type a name for the VOI in File name, and click Save. The program saves the file under the name you specified.



Note: When you are creating a script in active mode, the word *\$active* appears in commands on the Record New Script dialog box. For example, if you flip the image horizontally, the program adds “Flip \$active1 Y” (refer to the following example).

5 Save and close the image.

6 Select File > Save in the Record New Script dialog box when you are finished creating the script. The Save dialog box opens.

7 Type a name for the script in File name.

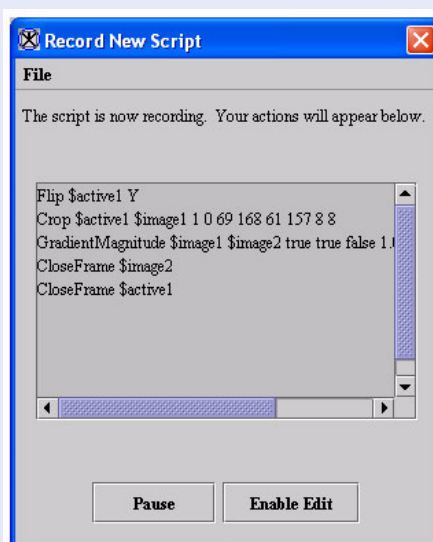
Tip: The file name extension for script files is .sct. Make sure that you type the extension at the end of the file name.

8 Click Save. The program saves the script in the scripts home directory.

9 Click File > Exit to close the Record New Script dialog box. The dialog box closes. You should now be able to select the script in Current Script.

An Example: Recording a Script in Active Mode

In this example, the user wanted to flip the image, delineate a VOI on the image, crop the image based on that VOI, and use the gradient magnitude algorithm on the cropped image. Notice the word *active*. The last two commands are closing the images.



To create scripts using the group mode

- 1 Start MIPAV. The initial MIPAV window and the Output window open.

Select File > Scripts > Record Script. The Record New Script dialog box (Figure 275A) opens. The following appears at the top of the dialog box: “The script is now recording. Your actions will be recorded below.”

- 2 Open an image on which you want to run the script. The MIPAV window expands.

The command **OpenImage \$image1** now appears in the scripting box (Figure 275B) in the Record New Script dialog box. If you continue to open images, the *\$imageN* attribute is sequentially ordered. That is, the command to open the second image is **OpenImage \$image2**; the command to open the third image is **OpenImage \$image3**; and so on.

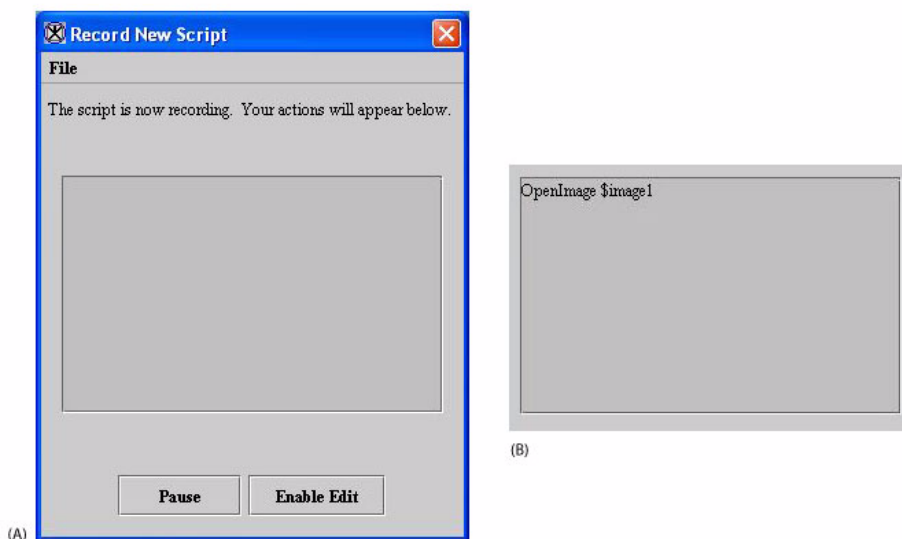



Figure 275. The (A) Record New Script dialog box and (B) the scripting dialog box after opening an image in group mode



Note: Just as when you’re creating a script in active mode, the Start recording script icon is red () to indicate that MIPAV is now recording a script.

- 3 Delineate VOIs and perform as many actions (algorithms and utilities) as you want on the image.

VOI commands do not appear on the scripting box on the Record New Script dialog box. To run the script later, you need to save the VOIs.

If you need to create a VOI to perform an action, save the VOI by selecting VOI > Save VOI. The program saves the VOI to the Scripts Home directory under the name *AreaN.voi* where *N* is a number that is assigned sequentially. The name of the first VOI that you save is *Area1.voi*; the second is *Area2.voi*; and so on. Make sure you type the .voi extension to the file.

If you want to specify a different name for the VOI, select VOI > Save VOI as. The Save VOI as dialog box opens. Type a name for the VOI in File name, and click Save. The program saves the file under the name you specified.

- 4** Close the images.
- 5** Select File > Save in the Record New Script dialog box when you are finished creating the script. The Save dialog box opens.
- 6** Type a name for the script in File name.

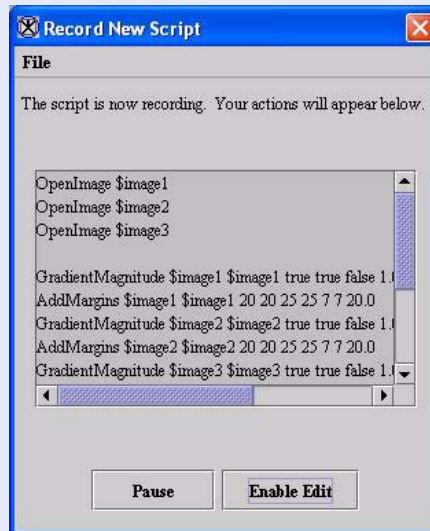


Tip: The file name extension for script files is .sct. Make sure that you type the extension at the end of the file name.



- 7** Click Save. The program saves the script in the scripts home directory.
- 8** Click File > Exit to close the Record New Script dialog box. The dialog box closes. You should now be able to select the script in Current Script.

An Example: Recording a Script in Group Mode

In this example, the user wanted to open three images, run the gradient magnitude algorithm on them, add margins on each image, and then close the images.



To stop recording scripts during creation

If you are interrupted or need to take a break while you are recording a script, the Pause button, , allows you to stop recording the script. When you click Pause, the following message appears at the top of the Record New Script dialog box: “The script is now paused. Press Resume to resume recording.” Notice that the Pause button changed to Resume, . When you can return to recording the script, click Resume.

Running scripts


Depending on your preference, you can run an active script that you just created from the scripting toolbar in the MIPAV window. You can also run scripts on multiple images from the File menu in group mode.

To run scripts from the scripting toolbar (active mode)

- 1 Select a script from Current Script in the scripting toolbar.



Note: Current Script lists all of the scripts in the scripts home directory alphabetically.

- 2 Click , the Run the selected script icon, on the scripting toolbar in the MIPAV window.

MIPAV automatically performs all of the actions in the script on the images indicated in the script.

To run scripts from the File menu (group mode)

- 1 Select File > Scripts > Run Script in the MIPAV window.

The Run Script on Multiple Images dialog box (Figure 276) opens.

- 2 Click Load Scripts. The Open dialog box appears showing all of the scripts in the scripts home directory.
- 3 Select a script.
- 4 Click Open.
- 5 Type, as an option, the prefix that should appear in front of the file name.

Load script	Allows you to select the script that you want to run from the scripts home directory. When you select this button, the Open dialog box appears.
Add prefix to saved image file name	Specifies an alphanumeric prefix that you want to apply to the images produced by the script.
Add suffix to saved image file name	Specifies an alphanumeric suffix that you want to apply to the images produced by the script.
Cancel	Disregards any changes that you made in the dialog box and closes this dialog box.
Back	Returns to the previous dialog box without running the script.
Next	Proceeds to the next step in running the script.
Finish	Begins running the script.

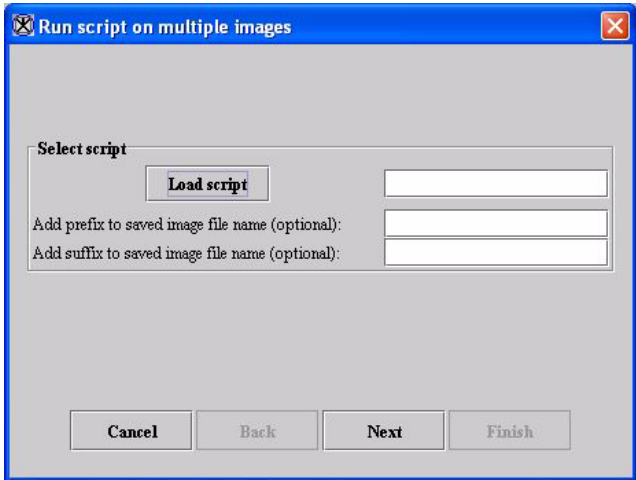


Figure 276. Run Script on Multiple Images dialog box

- 6** Type, as an option, the suffix that should appear at the end of every file name.
- 7** Click Next. The Run Script on Multiple Images dialog box that allows you to select images and VOIs appears.
- 8** Click Add under Images to add the images that you want to run the script on. The Open dialog box appears.

Images	Lists the images that appear in the script.
Add	Allows you to add one or more images on which to run the script.
Remove	Allows you to remove one or more images from those on which you plan to run the script.
VOIs	Lists any saved VOIs that are necessary when you run the script.
Add	Allows you to add one or more VOIs that the script needs to run correctly.
Remove	Allows you to remove one or more VOIs from use with the script.
Cancel	Disregards any changes that you made in the dialog box and closes this dialog box.
Back	Returns to the previous dialog box without running the script.
Next	Proceeds to the next step in running the script.
Finish	Begins running the script.

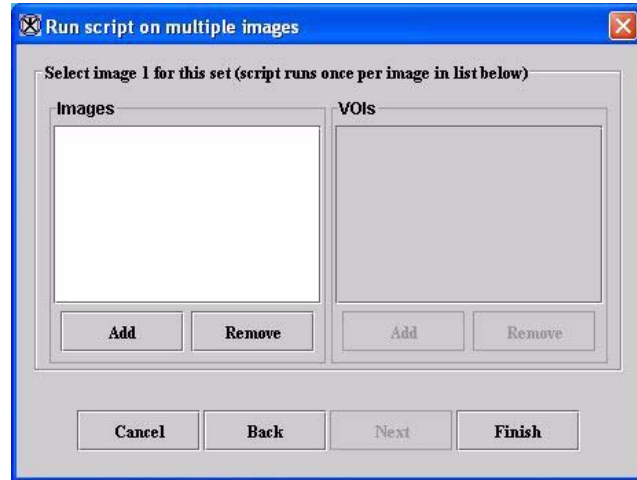


Figure 277. Run Script on Multiple Images dialog box (image and VOI selection)

- 9** Select one or more images, and click Open. The selected images appear in the Images box.
- 10** Select Add under VOIs to add one or more VOIs. The Open dialog box appears.
- 11** Select one or more VOIs, and click Open. The selected VOIs appear in the VOIs box.
- 12** Click Finish.



MIPAV automatically performs all of the actions in the script on the images indicated in the script.

Editing and deleting scripts

Editing scripts is very easy. You can either edit the script while you are creating it, or you can edit a previously saved script. MIPAV saves the script in text format so that any text editor, such as Microsoft WordPad or NotePad, can modify the file.

To edit scripts during creation

Suppose you make an error while you are recording a script, or perhaps you change your mind about performing a specific action. It's more practical to correct the problem right away.

To correct errors or make changes to scripts, simply click , the Enable Edit button, in the Record New Script dialog box (Figure 278). Two things occur: (1) the list box in the dialog box turns from gray to white to indicate that you can now type, copy, or paste information into it and delete information from it and (2) the Enable Edit button changes to , the Disable Edit button. If the box contains actions that you want to erase, select the actions and delete them. If you want to add new actions, you can either type them into the box directly, or you can record them after you finish editing the script.

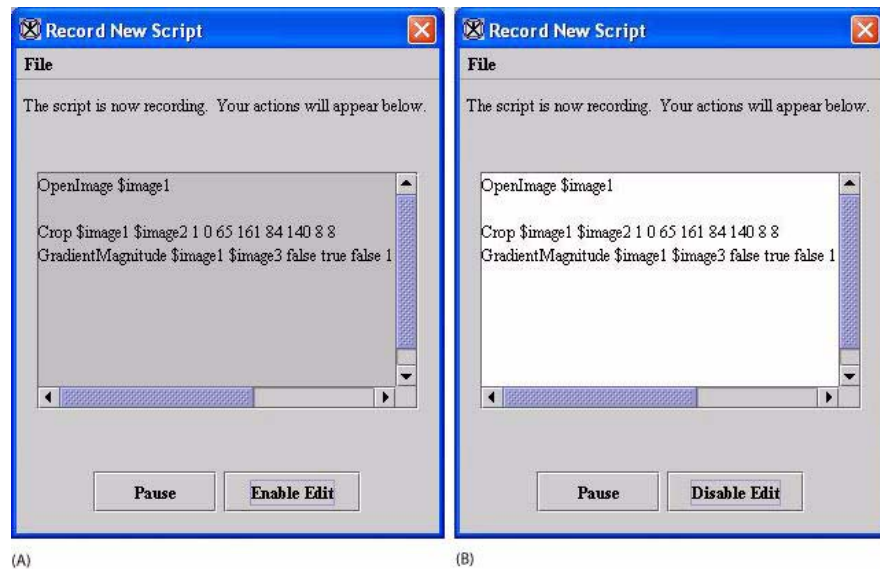



Figure 278. Record New Script dialog box (A) before and (B) after clicking Enable Edit

When you are finished changing the script, click Disable Edit. The list box turns gray and the Disable Edit button becomes the Enable Edit button.

To edit previously created scripts

- 1** Click , the Start Recording Script icon, on the scripting toolbar. The icon turns red, and the Record New Script dialog box opens.
- 2** Select File > Open. The Open dialog box (Figure 279) appears.
- 3** Select the script that you want to edit. The script appears in the Record New Script dialog box.
- 4** Click Enable Edit. The actions box turns from gray to white.
- 5** Select any of the actions and copy, rearrange or delete actions from the box or you can type new actions directly into the box.
- 6** Click Disable Edit when you are satisfied with the script. The actions box returns to gray.
- 7** Click File > Save if you are satisfied with the script. The Save dialog box opens.

- 8** Type the name of the script in the File Name box.
- 9** Click Save. MIPAV saves the script under the name you specified.

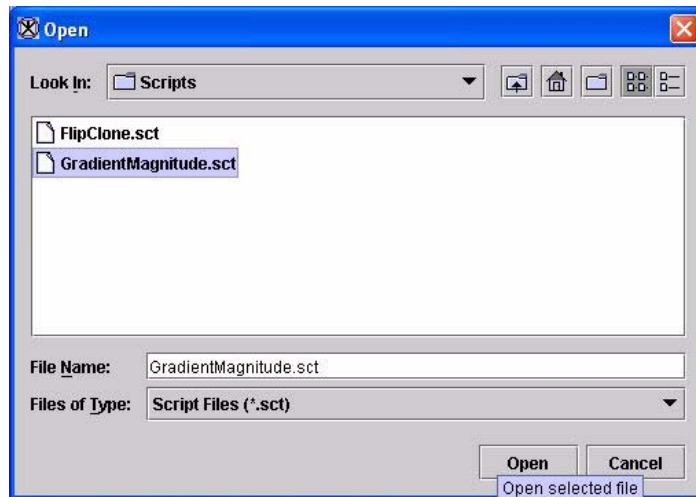


Figure 279. Open dialog box showing scripts in the scripts home directory

To delete scripts

If you decide at some point that you want to delete a script, go to Windows Explorer and navigate to the scripts home directory.

- 1** Right-click Start on the Windows desktop.
- 2** Select Explore. The Windows explorer window opens.
- 3** Navigate to the scripts home directory.
- 4** Select the script or scripts that you want to delete.
- 5** Right click.
- 6** Click Delete. Windows deletes the scripts you selected and places them in the Recycle Bin where they stay until you empty the bin.

Combining scripts and other programs

To increase productivity and efficiency, you can integrate MIPAV functions into your normal workflow by calling the scripts you've created. A simple example is creating a DOS batch file that opens and runs MIPAV scripts.

Using the mipav command

To call scripts from other programs, you use the **mipav** command in the Command Prompt dialog box. The correct syntax of this command follows.

Syntax of the mipav command

```
mipav [-hH] [-iI] imageFileName [-sS] ScriptFileName [-vV] voiFileName [-hideHide]
```

Parameters Purpose

[-h][-H]	Displays help for the mipav command in a Command Prompt window
[-hide][-HIDE]	Hides application frame
[-i][-I]	Image file name
[-s][-S]	Script file name
[-x][-X]	XML script file name
[-v][-V]	VOI file name

Examples

```
> mipav

> mipav imageFileName

> mipav -i imageFileName -s scriptFileName -hide

> mipav -s scriptFileName -i imageFileName1 -v voiName1 -v voiName2 -i
imageFileName2 -v voiName3
```

To display help for using the mipav command

- 1** Navigate to the mipav directory on your computer.
- 2** Select Start > All Programs > Accessories > Command Prompt. The Command Prompt dialog box opens.
- 3** Type **mipav -H** (refer to Figure 280).

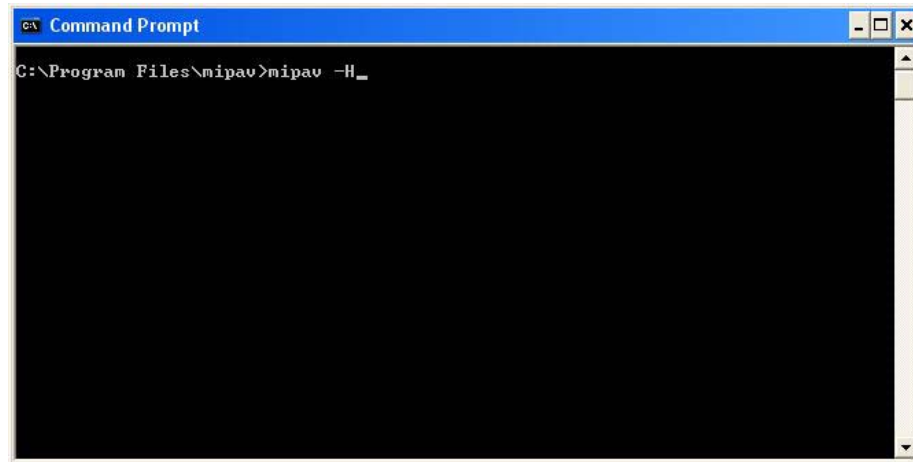


Figure 280. Command Prompt dialog box showing command to open Command Line Help dialog box

- 4** Press Enter. The Command Line Help dialog box (Figure 281) opens.

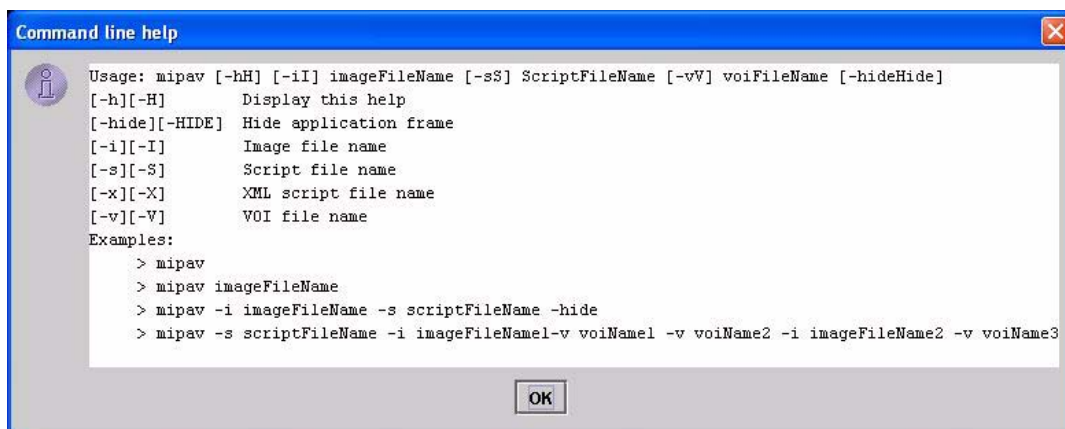


Figure 281. Command Line Help dialog box, which shows the syntax of the mipav command as well as examples

To open a DICOM image dataset

Suppose you want to open a single DICOM image from a collection of experiments made in 2004 named exp2004. You would type the following command in the Command Prompt dialog box in Microsoft Windows XP:

```
C:\ mipav -i i:\images\DICOM\exp2004\I04301.dcm
```

To open VOIs into that image

You can open VOIs as well as the image from the command line. In Windows XP, it would be the following:

```
C> mipav -i i:\images\DICOM\exp2004\I04301.dcm -v i:\VOIs\exp2004\levelset1.xml
```

In a UNIX BASH shell, this command looks like:

```
$ mipav -i ~/images/DICOM/exp2004/I043401.dcm -v ~/VOIs/exp2004/levelset1.xml
```

To open multiple images using compound commands

Suppose you know that there were multiple DICOM datasets in exp2004. To open every DICOM image on the Windows computer, you would type:

```
C> for %f in (i:\images\DICOM\exp2004\*01.dcm) do mipav -i %f
```

In this case, you must know something of the file structure of that dataset—you assumed that all image datasets had only one image ending in 01. However, the disadvantages of this format is the possibility of not opening all of the images at the same time.

A similar loop to open image sets on a UNIX BASH command line looks like:

```
$ for FI in `ls ~/images/DICOM/exp200?/*01.dcm`; do ./mipav -i $FI &; done
```

There are three significant differences between the BASH command and the Windows command (beside from how a directory is specified):

- **The use of the `ls` command when listing the directory**—The reason you must list the `ls` is due to the way a `for` loop works in BASH. The **`for`** requires a command and uses that command's return value as the boolean test to continue repeating the interior list of commands. By contrast, the Windows command shell expects a list of files. So long as the file listing has more results to list, BASH continues to repeat the **`mipav`** command.
- **The use of a wildcard when listing the directory**—BASH allows the directory list to use wildcard characters in more than one location, which permits searching for the images in any seven-character directory beginning with `exp200` as well as all files ending in `01.dcm`. This means that MIPAV starts with images from the `exp2004` directory, as well as `exp2003` or should it exist, `exp200M`, since the `?` matches any character, not just a number. This is an example of a feature of the shell being used to expand the results. Windows command shell does not support this feature.
- **Sending the `mipav` command to operate in the background**—BASH is a shell that allows *job control*. Using this feature allows you to start MIPAV and continue it asynchronously, permitting BASH to retain control. BASH can then continue processing the loop and starting MIPAV with the next matching file. Each MIPAV runs concurrently and allows you to manipulate each image with MIPAV at will. Although this allows you to see the images at the same time, the disadvantage is that the various windows begin to clutter the screen causing operator confusion.

When there is more than one MIPAV application window running, it's possible to close the wrong image by closing the wrong MIPAV application. In addition, operations that can occur between windows when running a single MIPAV may not be transferable between images being run by separate MIPAV windows.

While starting more than one MIPAV to display a set of images may be fine in limited applications, it causes needless overhead within in the operating system wasting system resources.

Using Shell scripting to lessen typing

Using shell scripts to reduce the amount of repetitive work is a common reason for writing a script. When best used, several small scripts that work

in concert can reduce the amount of typing required and the amount of time needed and can automate tasks.

The following example uses a Windows command shell to illustrate how you can shorten the number of keystrokes required. In this case, you would write a batch file to load a levelset VOI into an image.

```
@echo off
rem -- %1 is the full path to the image file, though not
rem -- the file itself; we assume there to be a *01.dcm
rem -- file to exist in this directory.

rem -- VOI is assumed to be in the same directory with name
rem -- levelset1.xml

./mipav -i %1\*01.dcm -v %1\levelset1.xml
```

More efficient and more useful, starting MIPAV with multiple images is easily done in a simple script. Here is how it is done in BASH:

```
#!/bin/bash
# argument 1 is the file (with wild-cards) we want to open
# arg 1 must be escaped (with quotes) to allow the shell to send
# the wildcards unmolested to the script. Otherwise, the shell
# will try to expand the shorthand. This has a different effect.

LISTING=`ls $1` # Generate the file listing
MIPARGS=
# For each file in the listing, prepend it with '-i' and
# the filename, then follow it with all the previous
# files.
for FS in $LISTING;
do
MIPARGS=" -i $FS $MIPARGS";
done

# start MIPAV:
./mipav $MIPARGS
```

Although this script doesn't include the line `./mipav $MIPARGS` with a "&" to run MIPAV in the background, it could have. This would have the effect of exiting the script with MIPAV in the background; as it is, the script does not exit—and return control to you at the command line—until MIPAV exits.